ยืน ภู่วรวรรณ

เทคนิคพัฒนาการ ภาษาไทย บนเครื่องไมโครคอมพิวเตอร์

น 2

คราวที่แล้วได้กล่าวถึงเรื่องเกี่ยวกับ บภาษาไทยอย่างกว้างๆ จนมาสิ้นสุดที่ ของโครงสร้างของ TSX บนเอ็มเอสดอส ปก็จะกล่าวถึงเรื่องที่เหลือเพื่อให้บทความ สมบูรณ์แบบ ก็ขอเชิญท่านผู้อ่านติดตาม ได้แล้วครับ

บรรทัด- 12 บรรทัด

บนบอร์ดแอปเปิ้ล 80 คอลัมน์ การ งผลตัวอักษรจะใช้ความละเอียดของจุด 7×9 จุด แต่ในไอบีเอ็มพีซี ถ้าใช้กับ มโนโครมแล้วเราจะได้ความละเอียด จุดในแต่ละตัวอักษรได้ถึง 9×14 จุด ถ้าใช้แบบจอสีจะได้ความละเอียดของตัว 8×8 จุด

การแสดงผลด้วยระบบบล็อกโมเสด พื่อการทำงานกับภาษาไทยนี้ จึงต้อง บรรทัดต่อการแสดงผล 1 บรรทัด ดังนั้น จอภาพจะแสดงภาษาไทยแบบถูกอักขระ ียง 8 บรรทัด

อย่างไรก็ตาม ได้มีการย่นย่อใช้ภาษา สองระดับ ดังรูปที่ 1

โครงสร้างการแสดงแบบนี้ จะรวม ล่างและสระบน(วรรณยุกต์ด้วย) เข้า วัน โดยในบางระบบ ถ้าสระบนพบสระ ะซ้อนทับกัน บางระบบจะต้องเตรียม สร้างตัวอักษรบนรอมตัวกำเนิดอักษร โดยมีกลุ่มรวมของสระล่าง เช่น ค .เป็นต้น ด้วยวิธีนี้จะทำให้จำนวนตัว รที่ใช้รวมตัวผสมแล้วมีประมาณ 250 ซึ่งต้องหาพื้นที่ของการสร้างตัวอักษร นี้ใน ROM ให้

อย่างไรก็ดี แนวทางของการพัฒนาจะ นกับฮาร์ดแวร์และรหัส ปัญหาในขณะนี้ รหัสที่ใช้มักมีขอบเขตจำกัดและเมื่อใช้ ภาษาไทย รหัสกราฟิกก็ใช้ไม่ได้

ภาษาไทย 25 บรรทัด

เทคโนโลยีการสร้างภาษาไทยเดิมมาจากบล็อกแสดงตัวอักษรขนาด 80×25 และเพื่อให้เห็นความแตกต่างของระบบ 25 บรรทัดกับแบบ 8–12 บรรทัด ซึ่งในส่วนของการรับคีย์และพิมพ์ผลลัพธ์ทางเครื่องพิมพ์จะเหมือนกัน สิ่งที่แตกต่างกันมากคือการแสดงผลออกมาทางจอภาพนั่นเอง และเพื่อให้เข้าใจระบบพื้นฐานได้ดีขึ้น จึงขอแสดงการกำหนดภาพบนจอของเครื่องไอบีเอ็มพีซีกันก่อน รูปที่ 2 เป็นการแสดงการเกิดภาพบนจอ และรูปที่ 3 เป็นตัวอย่างของการแสดงผลแบบบล็อกโมเสด 9×14 จุด

ในกรณีจอภาพแบบโมโนโครม ความละเอียดของตัวอักษรมีมากกว่าแบบกราฟิกสี จอภาพโมโนโครมจึงให้ตัวอักษรที่คมชัด การอ้างอิงตำแหน่งจอภาพกับหน่วยความจำของจอโมโนโครมใช้ตำแหน่งแอดเดรสจาก B0000 เป็นต้นไป โดยไบท์คู่เป็นตำแหน่งของตัวอักษร และไบท์คี่เป็นตำแหน่งของแอตตริบิ้ว เช่น ถ้าต้องการกำหนดตัวอักษร ก ไปบนขอบซ้ายบนของจอภาพ จะใช้ด้วยเบสิกได้ดังนี้

```
10 DEF SEG &HB000
20 POKE 0, 161
```

หากต้องการกำหนดแอตตริบิ้ว เพื่อให้ตัวอักษรที่ปรากฏเป็นอย่างไร เราทำได้โดยกำหนดค่าให้ไบท์แอตตริบิ้ว โดยรูปแบบของแอตตริบิ้วเป็นดังรูปที่ 4

ในการกำหนดแอตตริบิ้วด้วยเบสิกนั้น เราจะใช้ค่าของแอตตริบิ้วด้วยเลขฐานสิบดังนี้

ATT = 128* (blink) + 16*(background) + Forground

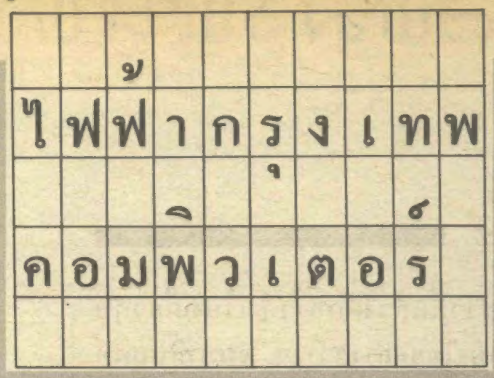
เช่น ถ้าให้ตัวหนังสือที่คอลัมน์บนสุดซ้ายมือ เป็นตัวหนังสือกะพริบโดยให้เป็นตัวรีเวิร์สด้วย เราจะใช้

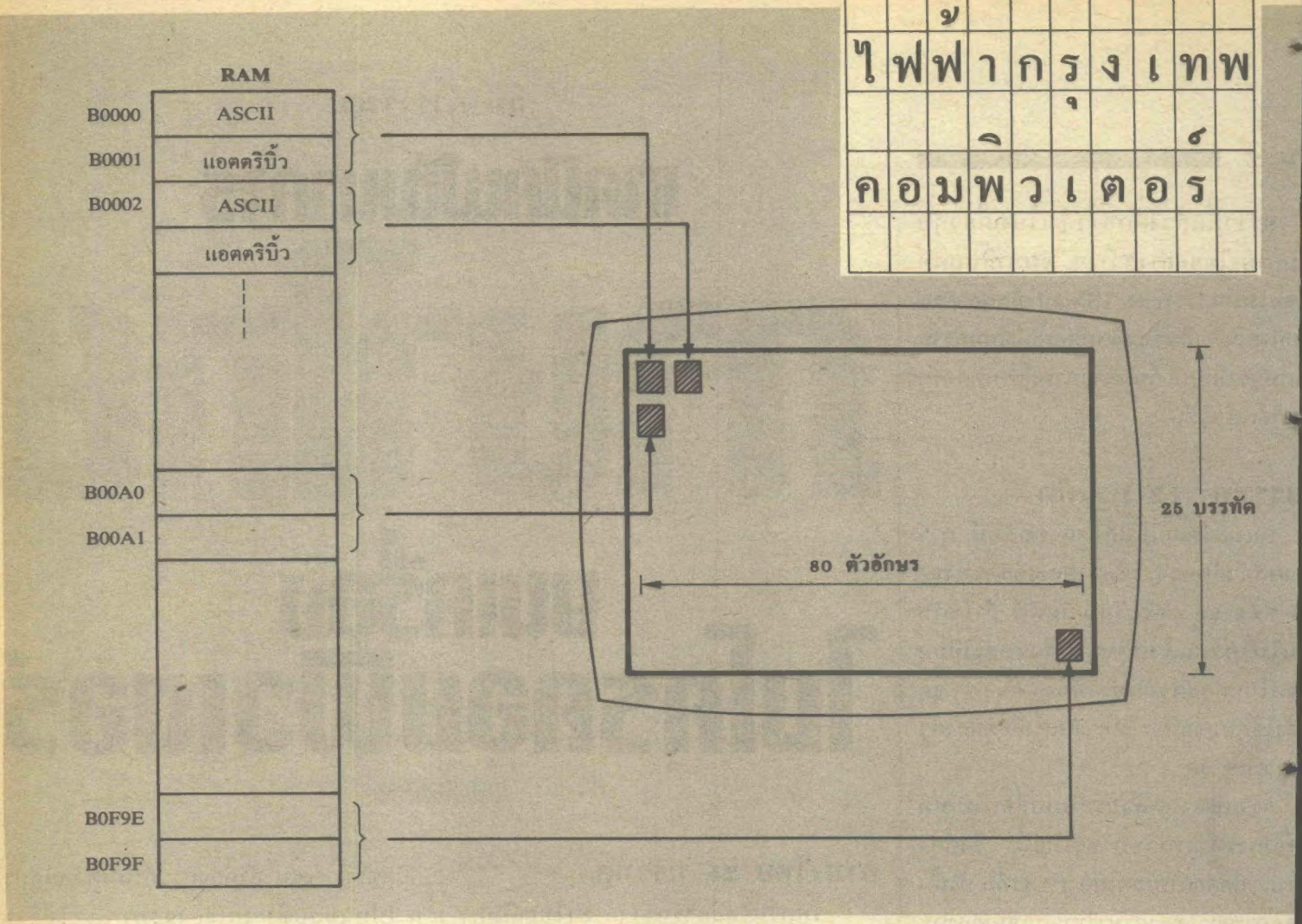
ATT = 128*(1) + 16*(7) + 8

= 248

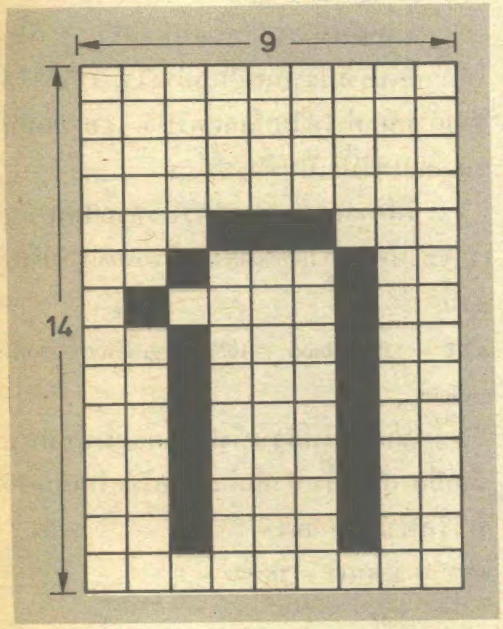
นั้นคือค่า blink = 1

รูปที่ 1 การแสดงผลแบบ 2 ระดับ

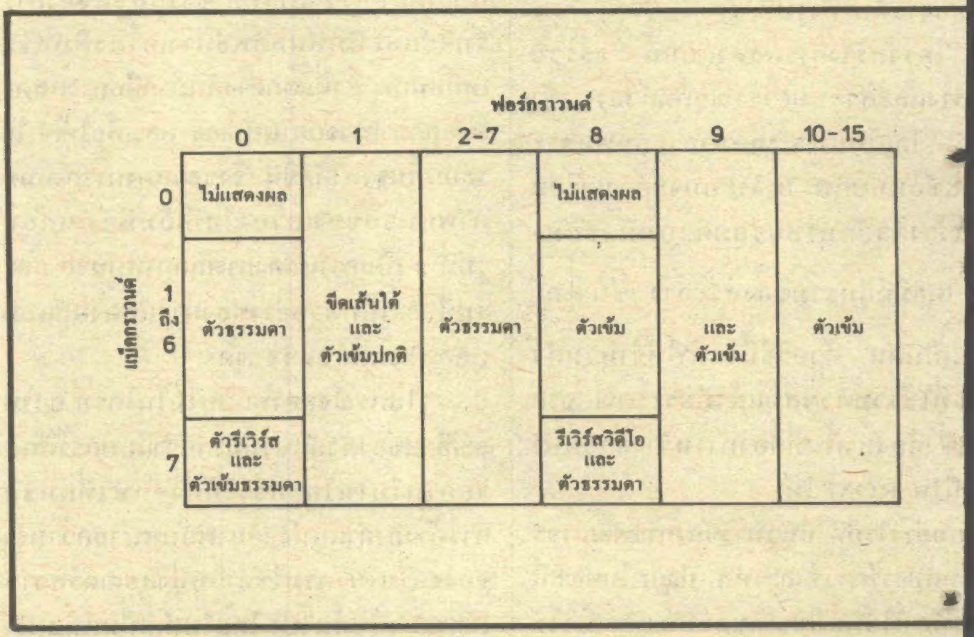




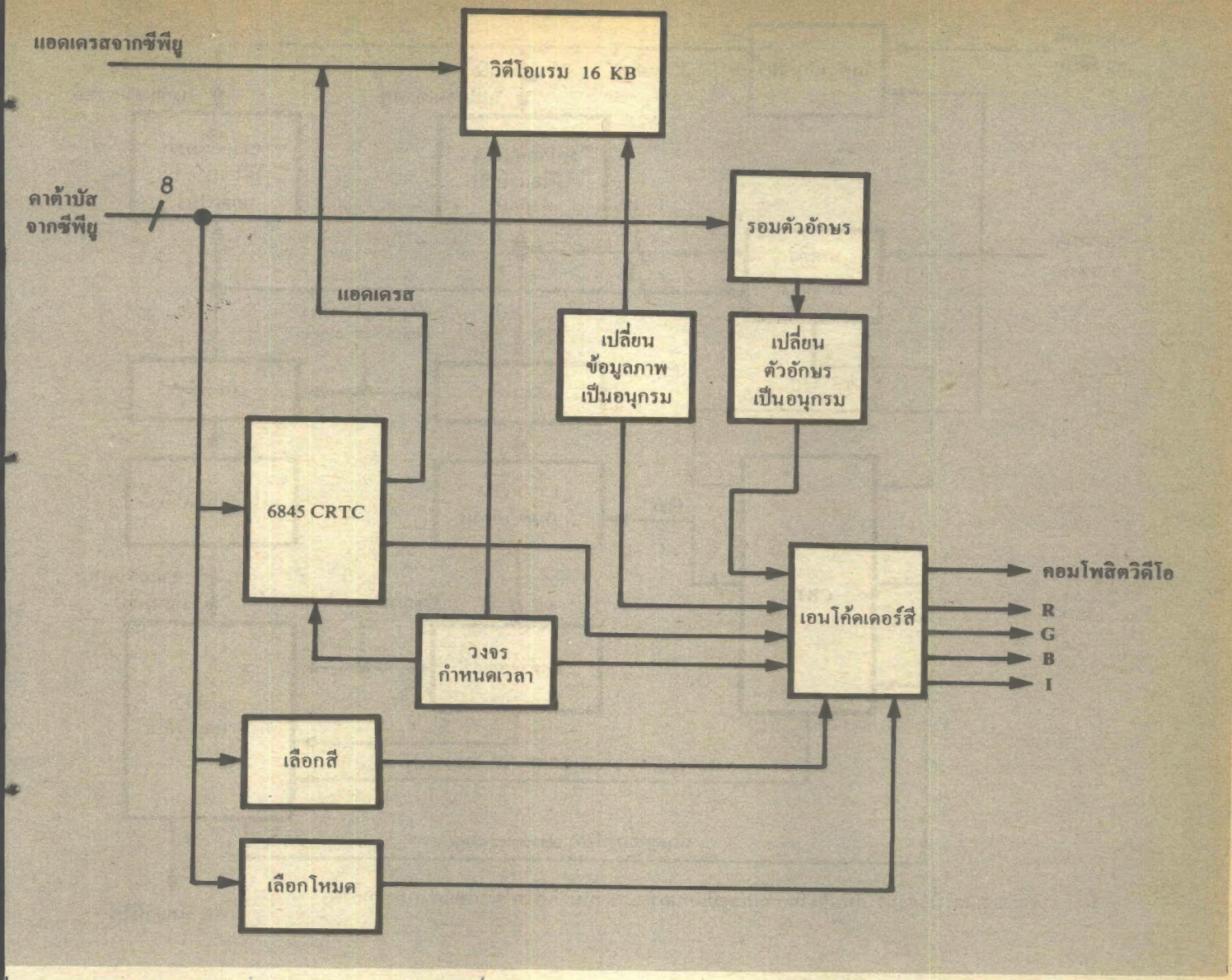
รูปที่ 2 แสดงการใช้หน่วยความจำกำหนดตัวอักษรบนจอโมโนโครมของ IBM-PC



รูปที่ 3 บล็อกโมเสดแสดง 9×14 จุด



รูปที่ 4 รูปแบบของแอตตริบิ้ว



รูปที่ 5 บล็อกไดอะแกรมของบอร์ดกราฟิก

background = 7

และ foreground = 8

ดังนั้น ถ้าใช้คำสั่ง

```
DEF SEG &HB000
POKE 0, 161
POKE 1, 248
```

จะได้อักษร ก เป็นตัวรีเวิร์สแบบกะพริบ

งสร้างฮาร์ดแวร์กับการทำภาษา
ย

บนเครื่อง 16 บิท จะใช้บอร์ดควบคุมยกต่างหาก ซึ่งมีให้เลือกหลายรูปแบบ

1. บอร์ดโมโนโครม
2. บอร์ดกราฟิกสี
3. บอร์ดโมโนโครมกราฟิก

เพื่อให้เห็นโครงสร้างการทำงานของฮาร์ดแวร์เพื่อเข้าใจหลักการสร้างภาษาไทยในระบบต่างๆ จึงใคร่ขออธิบายหลักการของฮาร์ดแวร์พอสังเขปดังนี้

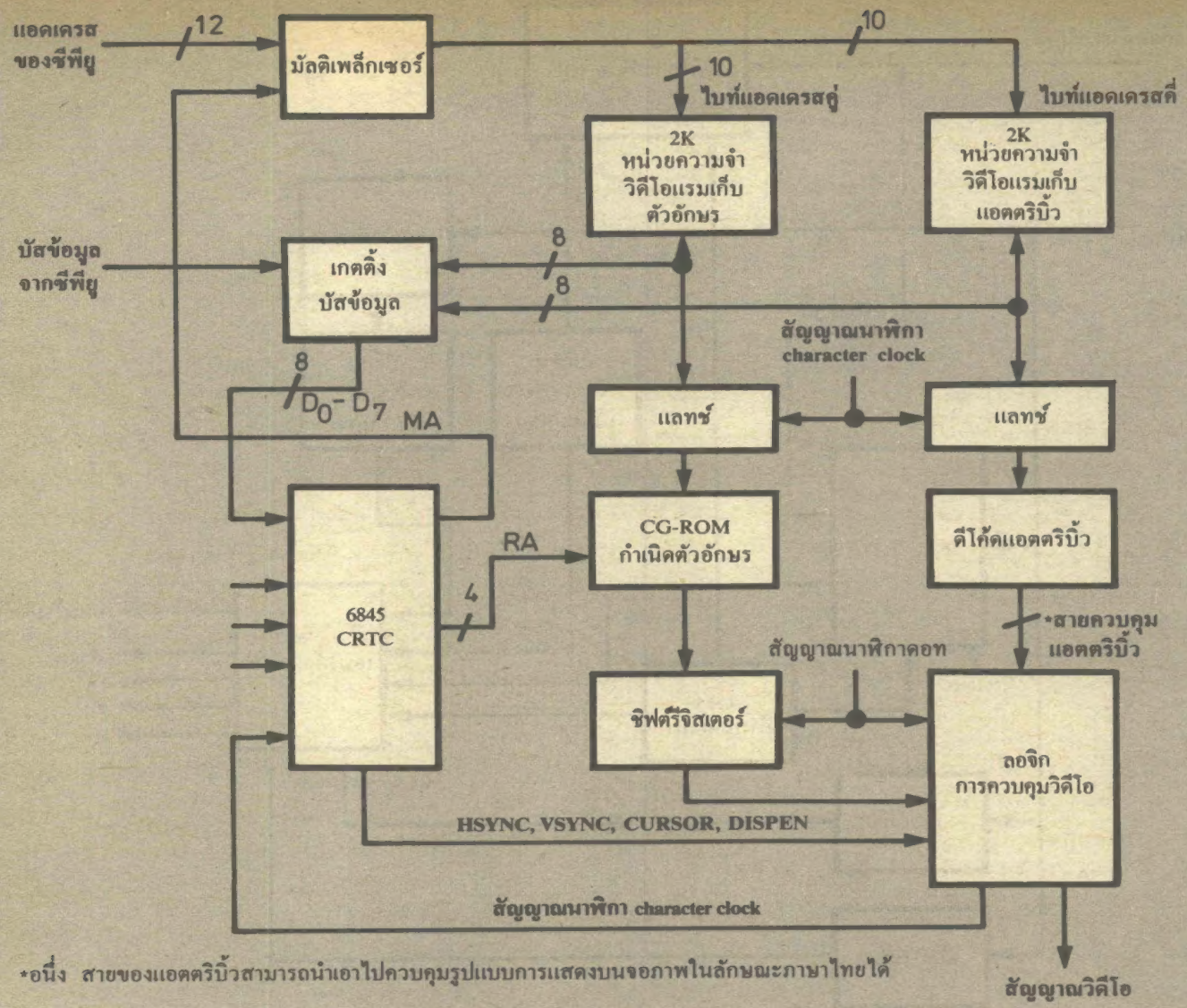
บล็อกไดอะแกรมตามรูปที่ 5 นี้ เป็นการกำหนดในรูปแบบของสีออกมาเป็นสัญญาณ R G B และ I โดยใช้แอตตริบิ้ว เป็นตัวสร้างสี คือส่วนการเลือกสี ในการกำหนดให้ตัวอักษรเป็นสีอะไร โดยการแปลงเป็นภาษาไทยแบบง่ายที่สุดคือการเปลี่ยนรอมตัวอักษรให้เป็นการแสดงภาษาไทยได้ด้วยการเปลี่ยน ROM จึงเป็นวิธีพื้นฐานในการสร้างภาษาไทยที่ใช้ในระบบ 8–12 บรรทัด ส่วนในรูปที่ 6 เป็นบล็อกไดอะแกรมของบอร์ดโมโนโครม

เทคนิคที่ใช้ในการทำภาษาไทยระบบ 24 บรรทัด

ในการทำภาษาไทยระบบ 24 บรรทัด มีเทคนิคการทำได้หลายรูปแบบ แต่ที่สำคัญคือจะต้องเกี่ยวข้องกับฮาร์ดแวร์อยู่พอควร ดังนั้นในบทความนี้จะสรุปเทคนิคต่างๆ พอสังเขป เพื่อเป็นแนวทางสำหรับการพิจารณาหรือปรับปรุงสำหรับอนาคต

1. 24 บรรทัดแบบกราฟิกล้วน

ไมโครคอมพิวเตอร์หลายเครื่องสามารถจัดการตัวอักษรในรูปแบบกราฟิก กล่าวคือสามารถจัดสร้างตัวอักษรในโหมดกราฟิกลงในแรม แล้วทำการรับตัวอักษรมาใส่ใน



รูปที่ 6 บล็อกไดอะแกรมของบอร์ดโมโนโครม

สกรีนตามตำแหน่งเคอร์เซอร์ การกำหนดรูปแบบของตัวอักษรนี้ จะต้องให้มีความละเอียดของจุดพอ ดังนั้นถ้าต้องการความกว้าง 80 ตัวอักษร และอักษรไทยแต่ละบรรทัดสูง 24 จุด (รวมสระบนล่าง) จะต้องใช้ความละเอียด 640 จุด × 600 จุด หรือหากทางสูงของจอให้ความละเอียดได้ 600 จุด แล้วจะได้ตัวหนังสืออย่างสวยงาม แต่ไมโครคอมพิวเตอร์ให้ความละเอียดทางด้านสูงของจอภาพเพียง 400 จุด ซึ่งอักษรไทย รวมสระบนล่างและวรรณยุกต์จะอยู่ในบล็อกที่ความสูงเพียง 16 จุด ซึ่งก็ทำให้ตัวอักษรสวยงามได้ยาก

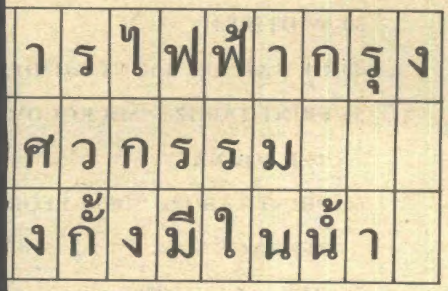
การดำเนินงานด้วยวิธีการทางกราฟิกนี้ จะต้องใช้ซอฟท์แวร์ช่วยในการจัดการ ซึ่งก็ต้องเสียเวลาในการทำงาน ดังนั้นโปรแกรมจัดการกราฟิกนี้จะใหญ่และจะทำให้การทำงานช้า อย่างไรก็ตามปัจจุบันมี CRTC บางเบอร์ เช่น 7220 สามารถจัดการรูปภาพได้ดีขึ้น การจัดการเคลื่อนย้ายบล็อกตัวอักษรจะทำได้รวดเร็วจนทำให้เทคนิคการแสดงผลแบบนี้เป็นไปได้ การจัดการกราฟิกนี้ ทำให้รูปแบบการทำงานสามารถเขียนตัวอักษรมา EX-OR หรือ IN-OR เพื่อนำวรรณยุกต์ซ้อนสระได้ ดังนั้นจึงไม่จำเป็นต้องเก็บรูปแบบตัวอักษรแบบผสมระหว่างสระกับวรรณยุกต์

อย่างไรก็ตาม โครงสร้างของ BIOS บางเครื่องสนับสนุนการเขียนอยู่บ้างแล้ว การเขียนโปรแกรมเพื่อให้ได้ผลจึงต้องใช้เทคนิคการเชื่อมต่อที่ดี ทางห้องปฏิบัติ
วิจัยไมโครเคยทดลองใช้ซอฟท์แวร์ชื่อ GSX ของดิจิตอลรีเสิร์ชช่วย ปรา
ได้ผลดี แต่สกรีนจะไม่เร็วมากนัก และ
สร้างรวมของ GSX กับ TSX จะกิน
ในหน่วยความจำมาก

2. การใช้รหัส 2 ไบท์

วิธีการนี้เป็นแนวความคิดของ
ต้องการเพิ่มความเร็วในการทำงาน เ
งานกับอุปกรณ์ I/O โดยเฉพาะเครื่อง
จะเห็นว่าเครื่องพิมพ์ปัจจุบันสามารถพิ
แนวตั้งได้ถึง 24 จุด การพิมพ์ภาษาไทย
ลากสแกนได้ทีละ 1 บรรทัด โดยพิมพ์วร
ยุกต์ พยัญชนะและสระล่าง พร้อมก
ทีเดียว ซึ่งก็เท่ากับ เพิ่มความเร็วในการ

า	ร	ไ	ฟ	ฟ้	า	ก	รุ	ง
ศ	ว	ก	ร	ร	ม			
ง	กั้	ง	มี	ใ	น	น้ำ		



7 รูปแบบการแสดงผลที่มีทั้งสระและวรรณยุกต์อยู่บรรทัดเดียวกัน

วามภาษาไทยได้อีกสามเท่าเชียว แต่กฏว่ายังไม่มีเครื่องพิมพ์ใดทำ

ความคิดนี้ สามารถนำมาประยุกต์กับาไทยได้ เพราะปกติของจอจะได้ 24 ทัดๆ ละ 80 ตัวอักษรอยู่แล้ว ดังนั้นถ้าะบรรทัดจะมีตัวอักษรอยู่ครบทั้งสระวรรณยุกต์ ก็จะทำให้การทำงานทำได้รีวมากยิ่งขึ้น โดยดูจากรูปแบบการแสดงามรูปที่ 7

การแสดงแบบนี้จะใช้บล็อกโมเสดตัวอักษรแต่ละตัวแสดงตัวอักษรทั้งสามบเลย ดังนั้นถ้าบล็อกตัวอักษรมีความยด 24 × 9 จุดแล้วก็จะแสดงได้สบาย เมื่อเป็นเช่นนี้จำนวนตัวอักษรที่จะแสดงกำเนิดอักษรจะต้องเพิ่มขึ้นเป็นทวีคูณมีอักษร ก ก่ ก้ ก๊ ก๋ กิ่ กิ้ ....กุ่ กุ้..และ

จากที่ห้องปฏิบัติการวิจัยไมโครบรวมตัวอักษรที่ใช้และขนาดของ ROM ปรากฏว่า ได้ทดลองทำได้โดยใช้ ROM ประมาณ 128 กิโลไบท์ในส่วนของคเตอร์เจนฯ โดยตัวอักษรทั้งหมดที่ใช้ประมาณ 2,800 ตัว

การกำหนดตัวอักษรจะใช้ 2 ไบท์ โดยหนึ่งเป็นไบท์นำบอกพยัญชนะตัวกลางบท์เสมือนเป็นแอตตริบิ้วที่จะบอกว่ามีบนล่างอยู่หรือไม่ และรูปแบบการกำหนดการแสดงผลของไอบีเอ็มก็ใช้ 2 ไบท์วจึงแมทช์กับการใช้ในกรณีนี้ได้อย่างดี ข้อดีของการกำหนด 2 ไบท์ต่อรหัสนี้แบบของการจัดการทางจอภาพ จะทำให้ฟิสิคัลโลเคชันของการแสดงผลของโปรแกรมสำเร็จรูปส่วนใหญ่จะใช้ไบท์ที่เป็นตัวอักษรและแอตตริบิ้วคู่กันอยู่แล้ว วิธีนี้จึงทำให้โลจิคัลและฟิสิคัลตรงกัน การแก้ไขซอฟท์แวร์จะง่ายกว่าแบบอื่น

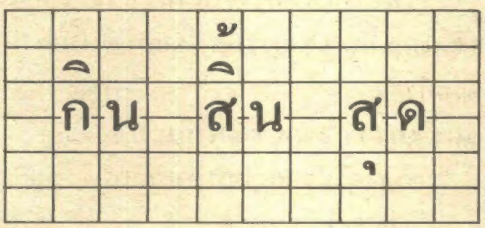
การใช้ 2 ไบท์แบบนี้จะสามารถทำให้มีรูปแบบของฮาร์ดแวร์ที่ใช้กับภาษาทางตะวันออกนี้ได้มาก เช่น ภาษาจีน ญี่ปุ่น ฯลฯ ดังนั้นเราอาจนำเอาซอฟท์แวร์ที่อาจจะพัฒนาขึ้นในละแวกนี้มาดัดแปลงได้ง่ายขึ้น การใช้ 2 ไบท์แบบนี้ สามารถดัดแปลงบอร์ดโมโนโครมหรือกราฟิกได้ง่าย โดยดึงบิทแอตตริบิ้วในส่วนของฮาร์ดแวร์มาถอดรหัสร่วมเพื่อเลือกตัวอักษรในแต่ละบล็อกโมเสดได้ โดยถ้ากำหนดกลุ่มการเลือก CG-ROM ดีๆ แล้วจะทำให้เขียนซอฟท์แวร์ในการผสมไบท์ทั้งสองได้ง่ายขึ้น ดังนั้นการจัดกลุ่มตัวอักษรจะช่วยทำให้ระบบฮาร์ดแวร์และซอฟท์แวร์ดูง่ายขึ้น

3. วิธีแบ่งส่วนตัวอักษร

วิธีนี้กำหนดรูปแบบของตัวอักษรเป็นส่วนย่อยขึ้น กล่าวคือกระจายบล็อกโมเสดมากขึ้น ลองดูจากรูปที่ 8

จะเห็นว่าในหนึ่งคอลัมน์ของอักษรไทยแบ่งเป็น 5 ส่วน คือ ส่วนวรรณยุกต์ ส่วนสระบน ตัวพยัญชนะส่วนบน ตัวพยัญชนะส่วนล่าง และสระล่าง เมื่อประกอบกันจะครบเป็นตัวอักษรไทย 1 บรรทัด ในกรณีนี้ CG-ROM จะเก็บรูปแบบของตัวอักษรเพียงแค่ 4×8 จุด (สูง×กว้าง) โดยส่วนพยัญชนะจะแยกเก็บเป็น 2 กลุ่มคือ กลุ่มส่วนบนกับกลุ่มส่วนล่าง เช่น ตัวอักษร ก จะประกอบด้วย ก และ ก เมื่อมาต่อกันจะเป็น ก

				้					
	ิ			ิ					
	ก	น		ส	น		ส	ด	
							ุ		



รูปที่ 8 การกระจายบล็อคโมเสด

ในการทำงานจะเห็นว่าถ้าให้แต่ละส่วนเป็น 4×8 ดังนั้นอักษรไทยหนึ่งคอลัมน์จะใช้ 20 จุด หรือจำนวนสแกนทั้งสิ้น 500 เส้น การแสดงภาพบนจอจะต้องใช้หน่วยความจำกำหนดตำแหน่งที่ยุ่งยากขึ้น โดยมีจำนวนหน่วยความจำ(video RAM) เพิ่มขึ้นเพื่อเก็บรูปแบบต่างๆ ได้ครบ ดังนั้นจึงต้องขยายวิดีโอแรมเพิ่มขึ้น ซึ่งผิดกับแบบ 2 ไบท์ที่ไม่ต้องขยายวิดีโอแรมเลย เพราะใช้แอตตริบิ้ว อย่างไรก็ตามการกำหนดตัวอักษรบนล่างก็ยังคงใช้บิทแอตตริบิ้วทางซอฟท์แวร์เข้าช่วย โดยให้แอตตริบิ้วไปสัมพันธ์กับส่วนของตำแหน่งตัวอักษร

วิธีการนี้จะใช้ฮาร์ดแวร์ที่ยุ่งยาก และมีส่วนของฮาร์ดแวร์เพิ่มเติมขึ้นมามาก ซึ่งจำเป็นต้องทำบอร์ดประกบอีกบอร์ดหนึ่ง วิธีการนี้จะยังคงใช้ซอฟท์แวร์ควบคุมที่ทำงานสลับซับซ้อน ส่วนของการจัดการส่วนนี้จะทำให้ระบบการแสดงผลช้าลงได้ แต่ทั้งนี้ต้องขึ้นกับเทคนิคของซอฟท์แวร์

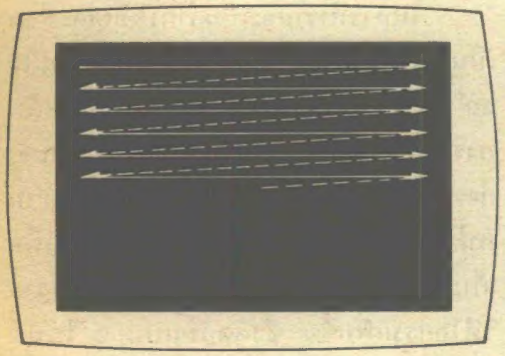
4. ใช้วิดีโอสกรีน 2 ซ้อน

ในบางเครื่องสามารถสร้างสกรีนกราฟิก สกรีนตัวอักษร หรือสร้างสกรีนภาพได้มากกว่าหนึ่งสกรีนมาซ้อนกัน เมื่อเป็นเช่นนี้จึงทำให้สแกนเส้นของภาพสูงขึ้น เช่น 400 เส้นหรือมากกว่า จากนั้นแสดงตัวหนังสือในระดับปกติบนสกรีนหนึ่ง และแสดงสระบนล่างอีกสกรีนหนึ่ง นำสัญญาณจากทั้งสองกรีนมาผสมกันเป็นสัญญาณควบคุมภาพ

วิธีการนี้จึงใช้จำนวน CG-ROM เท่ากับในระบบ 8–12 บรรทัด แต่จะต้องมีส่วนของฮาร์ดแวร์กำหนดการแสดงในแต่ละสกรีนและซอฟท์แวร์ควบคุมแต่ละสกรีนให้สัมพันธ์กัน

5. การใช้ interlace

ทุกแบบที่กล่าวมาแล้วจำเป็นต้องให้จำนวนเส้นของการสแกนสูง ทั้งนี้เพื่อที่จะให้บรรจุตัวอักษรที่อาจเพิ่มมากขึ้นเข้ามาบนจอภาพอีกถึงสามเท่าตัว



รูปที่ 9 ลักษณะการสแกนของจอภาพ

เพื่อเป็นพื้นฐานของการเข้าใจในการแสดงผลของจอภาพ ลักษณะการสแกนของจอภาพจะเป็นดังรูปที่ 9

บนบอร์ดโมโนโครมหรือโมโนโครมกราฟิก(เฮอร์คิวลิสต์) ใช้สัญญาณนาฬิกาจากคริสตอล 16.257 MHz ซึ่งก็หมายถึงสัญญาณนาฬิกากำหนดดอท(จุด) หรือจุดที่แสดงจะแสดงด้วยความเร็วถึง 16,257,000 จุดต่อวินาที แต่การแสดงของตัวอักษรใช้ 9 จุดในแนวนอน ดังนั้น 16,257,000/9 = 1,806,333 หรือเรียกว่า character clock สัญญาณนี้จะป้อนให้ 6845 (ตัวควบคุมการแสดงผล–CRTC) โดย CRTC จะนำรหัสตัวอักษรจากวิดีโอแรมส่งมาให้ ROM กำเนิดตัวอักษร เพื่อส่งข้อมูลตัวอักษรที่เป็นจุดสว่างมีดออกไป

หากพิจารณาเส้นในแนวนอนแต่ละเส้นจะเห็นว่าต้องแสดงตัวอักษร 80 ตัว แต่โดยความเป็นจริงแล้วต้องใช้เวลาในการสแกนเส้นกลับมาจุดเริ่มทางซ้ายมือใหม่ ดังนั้นแต่ละเส้นของการสแกน จะใช้เวลาเท่ากับการแสดงผล 98 ตัวอักษร (แสดงจริงเพียง 80 ตัวอักษร) หากวัดด้วยมาตรฐาน 98 ตัว การสแกนนี้จะเท่ากับจำนวนเส้นทั้งสิ้น 18,432 เส้นต่อวินาที ซึ่งจำนวนเส้นของการสแกนใน 1 วินาทีของจอโมโนโครมนี้สูงกว่าทีวีซึ่งทีวีแสดงเพียง 15,734 จะเห็นว่าการสแกนของโมโนโครมสูงกว่าทีวี และถ้าหากให้ความละเอียดสูงขึ้น จะต้องใช้สัญญาณดอทสูงขึ้น ความถี่สูงขึ้น ซึ่งต้องแก้ไขฮาร์ด–แวร์มากขึ้น

คราวนี้มาดูการสแกนเป็นเฟรม จอโมโนโครมแสดงผล 25 บรรทัด โดยแต่ละบรรทัดสแกนทั้งสิ้น 14 เส้น (ตัวอักษร 14×9 จุด) ดังนั้นบนจอภาพจึงเท่ากับ 25×14 = 350 เส้น แต่อย่างไรก็ตามจะต้องเสียเวลาไปประมาณ 20 เส้น สำหรับการรีเทรซจากการย้อนกลับเฟรม ดังนั้นจึงต้องเท่ากับ 370 เส้นต่อหน้าจอ และถ้าเรานำเอา 18,432/370 จะมีค่าเท่ากับ 49.82 หรือเท่ากับการแสดงเฟรมของภาพ 49.82 เฟรมในหนึ่งวินาที (ประมาณ 50 เฟรมต่อวินาที ตามสเปคที่กำหนด)

เราโปรแกรม CRTC(6845) ได้อย่างไร

ในการทำภาษาไทยนี้ เราสามารถให้จำนวนเส้นต่อเฟรมสูงขึ้น โดยตัว CRTC จะมีพอร์ทเอาท์พุทให้เราโปรแกรมค่าได้ การกำหนดค่าให้รีจิสเตอร์ของ 6845 โดยรีจิสเตอร์ใน 6845 มี 18 รีจิสเตอร์ โดยมีความหมายดังตารางในรูปที่ 10

ในบอร์ดกราฟิกสีของไอบีเอ็ม กำหนดพอร์ท กำหนดรีจิสเตอร์เป็น 3D0 และพอร์ทกำหนดค่าให้รีจิสเตอร์เป็นพอร์ท 3D1 เช่น โหมดอินเตอร์เลซ คือการลดการสแกนของเฟรมลงครึ่งหนึ่ง (ปกติแสดง 50 เฟรมต่อวินาที) โดยเราจะลดลงเหลือ 25 เฟรมต่อวินาที และให้จำนวนเส้นสูงขึ้นอีกเท่าตัว เราก็จะได้ตัวอักษรเป็น 50 บรรทัด และแสดงอักษรไทยแบบ 2 ระดับ ก็จะได้ภาษาไทย แบบ 25 บรรทัดได้

เพื่อให้เห็นการทำงานของจอสีที่จะแสดงผลเป็น 50 บรรทัด โดยการโปรแกรมด้วยคำสั่ง

```
OUT &H3D0, 8 :OUT &H3D1, 3
```

ลองดูโปรแกรมภาษาเบสิกนี้

```
10 SCREEN 0, 0
20 CLS
30 WIDTH 80
40 OUT &H3D0, 8 :OUT &H3D1,
50 PRINT TAB(12); "MICROCOM
   TER JOUNAL"
60 PRINT TAB(12); "DEMO FOR
   TERLACE"
70 DEF SEG &HB800
80 FOR I = 3 TO 49
90 FOR J = 0 TO 79
100 POKE 160*I + 2*J, 161
110 NEXT I
120 NEXT J
```

ลองรันดูจะได้ตัวอักษรเล็กๆ ขอ
ก เต็มจอ ซึ่งเป็นวิธีการทำแบบ 25 บร
ไทย โดยไม่ต้องปรับฮาร์ดแวร์

อนึ่งการโปรแกรมอินเตอร์เลซนี้
เป็น 0 หรือ 2 คือโปรแกรมให้
แบบนอนอินเตอร์เลซ แต่ถ้าเป็น 1 จะท
เกิดการซ้ำจอครึ่งบนล่าง แต่ถ้าเป็น 3
แยกออกจากกัน

แต่สำหรับจอโมโนโครมเราลอง
จากโปรแกรมต่อไปนี้ โดยกำหนดพ
สำหรับ 6845 เป็น 3B4 และ 3B5 (กำ
ด้วยฮาร์ดแวร์ของไอบีเอ็ม)

```
10 KEY OFF
20 CLS
30 WIDTH 80
40 OUT &H3B4, 8 :OUT &H3B5,
50 OUT &H3B4, 1 :OUT &H3B5,
60 OUT &H3B4, 2 :OUT &H3B5,
```

เสร็จแล้วลองใช้

```
65 PRINT "THAI 25 LINE"
70 DEF SEG = &HB000
80 FOR I = 3 TO 49
90 FOR J = 0 TO 39
100 POKE 80*I + 2*J, 161
110 NEXT J
120 NEXT I
```

ลองใช้คำสั่ง RUN จากนั้น ล

10 ตารางแสดงความหมายของรีจิสเตอร์ต่างๆใน 6845

รีจิสเตอร์	ความหมาย	ค่าเริ่มของแผ่น Graphic Adapter			โมโนโครมอะแดปเตอร์
		40 Col Text	80 Col Text	Graphic	
0	ช่วงเวลาแนวแกนราบทั้งหมด	56	113	56	97
1	จำนวนตัวอักษรในแนวแกนราบ	40	80	40	80
2	ตำแหน่งการซิงโครไนซ์แกนราบ	45	90	45	82
3	ความกว้างของซิงโครไนซ์แกนราบ	10	10	10	15
4	จำนวนเส้นในแนวดิ่งทั้งหมด	31	31	127	25
5	การปรับค่าแนวดิ่ง	6	6	6	6
6	จำนวนเส้นที่แสดงในแนวดิ่ง	25	25	100	25
7	ตำแหน่งการซิงโครไนซ์แนวดิ่ง	28	28	112	25
8	อินเตอร์เลซโหมด	2	2	2	2
9	จำนวนเส้นสแกนสูงสุด	7	7	1	13
10	เส้นเคอร์เซอร์เริ่มต้น	6	6	6	11
11	เส้นเคอร์เซอร์สิ้นสุด	7	7	7	12
12*	แอดเดรสหน่วยความจำเริ่มต้น (high)	0	0	0	0
13*	แอดเดรสหน่วยความจำเริ่มต้น (low)	0	0	0	0
14*	แอดเดรสของเคอร์เซอร์ (high)	0	0	0	0
15*	แอดเดรสของเคอร์เซอร์ (low)	0	0	0	0
16*	ตำแหน่งปากกาแสง (high)	—	—	—	—
17*	ตำแหน่งปากกาแสง (low)	—	—	—	—

* สามารถเขียนและอ่านดูได้

จะเห็นว่าจอภาพจะเป็นตัวอักษรเล็ก รรทัดๆ ละ 40 ตัวอักษร

ความหมายของโปรแกรมคือ กำหนด นเตอร์เลซ จำนวนตัวอักษรต่อบรรทัด ว และเริ่มแสดงตัวแรกในตำแหน่ง 65 ห้อยู่กลางจอ

หากพิจารณาดูจะเห็นว่าตัวอย่างนี้ ดไว้เพียงบรรทัดละ 40 ตัวอักษรเพราะ แรม มี 4K ซึ่ง 2K เป็นที่เก็บตัวอักษร ก 2K เป็นแอตตริบิ้ว ดังนั้นจึงแสดง ง 80×25 หรือ 40×50 นั่นเอง ดังนั้น ต้องการให้ใช้ไทย 25 บรรทัดแบบ กับ ต้องขยายหน่วยความจำวิดีโอแรม ตัวก็จะแสดงได้โดยปกติ เราก็ทำได้ ารเพิ่มบอร์ดขยายเข้าไป

อย่างไรก็ดีมีบอร์ดเฮอร์คิวลิสต์ ซึ่ง บบมามีวิดีโอแรมสูงถึง 64 KB ซึ่ง ายเหลือเฟือ แต่เป็นการออกแบบที่มีจุดปัญหาสำหรับภาษาไทยอยู่อย่างหนึ่ง คือ เรื่องของแอตตริบิ้วซึ่งถ้าวิเคราะห์วงจรแล้วจะพบจุดนี้ แต่ก็มีทางแก้ได้ไม่ยาก ดังนั้นจึงมีผู้พัฒนาบนบอร์ดนี้กันมากในระยะหลัง เพราะเป็นแนวทางที่จะทำได้ผลดีในเชิงฮาร์ดแวร์ แต่อย่างไรก็ตามบอร์ดโมโนโครมที่ขายในเมืองไทยมีประมาณ 5 แบบ แต่ละแบบแตกต่างกันในรายละเอียดดังนั้นวิธีการดัดแปลงจึงแตกต่างกันหมด

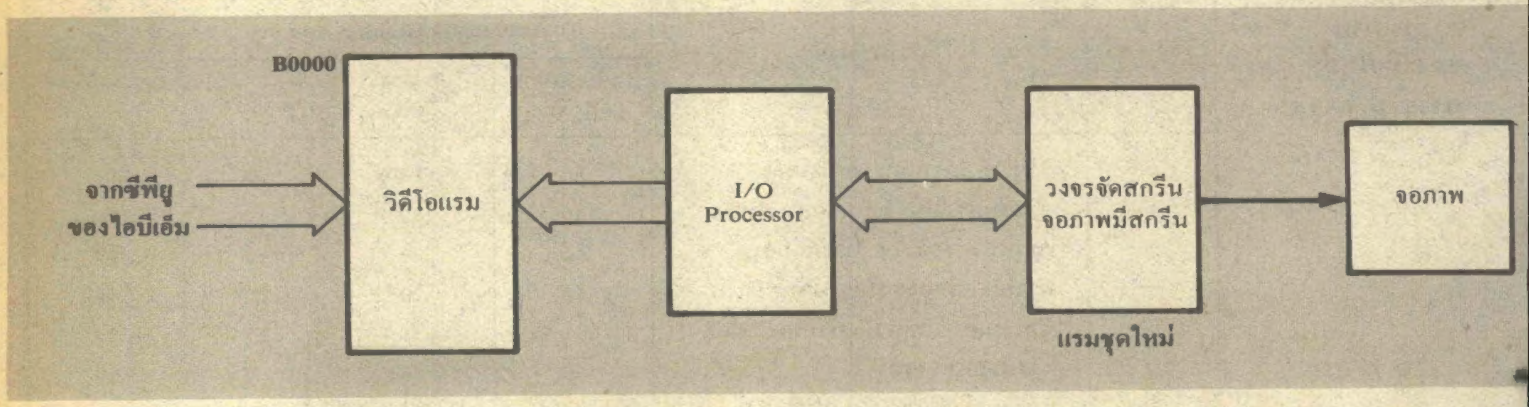
เมื่อขยายวิดีโอแรมไว้แล้วเราก็จะต้องบอก DOS ให้รู้ว่ามี 50 บรรทัดแล้ว การบอก DOS นี้ทำได้ไม่ยากมีการทำได้หลายแบบ แต่ถ้าทำในดีไวซ์ไดรเวอร์แล้ว จะดีเพราะไม่ต้องยุ่งกับ BIOS การจัดการด้วยวิธีนี้จะสามารถใช้กับ Thai.com ของห้องปฏิบัติการได้ แต่การขยายเป็น 50 บรรทัดต้องแก้ Thai.com อีกเล็กน้อยเท่านั้น ระบบ 25 บรรทัดก็จะใช้ได้

ปัญหาใหญ่ที่ตามมาคือ โปรแกรมสำเร็จรูปบางตัวยังคงมีปัญหากับ TSX ดังนั้นต้องแก้ไขเป็นพิเศษ

6. การใช้ IOP(Input-Output Processor)

วิธีการนี้เท่าที่สังเกตในตลาดยังไม่มีผู้ใดใช้เพราะจะต้องออกแบบฮาร์ดแวร์เองทั้งหมด แต่จะเป็นแบบที่แก้ปัญหาในเรื่องโปรแกรมสำเร็จรูปได้ โครงสร้างของฮาร์ดแวร์จะใช้ไมโครโปรเซสเซอร์ทำการอ่านวิดีโอแรมของไอบีเอ็มมาจัดการภาพใหม่ตลอดเวลา โดยส่วนของวิดีโอแรมของไอบีเอ็มจะทำเป็นแบบสองพอร์ท โดย IOP จะอ่านมาจัดระดับแล้วแสดงใหม่ วิธีการนี้จะใช้ได้อย่างกว้างขวางกับซอฟท์แวร์ทุกชนิด และไม่จำเป็นต้องดัดแปลงซอฟท์แวร์สำเร็จรูปเลย ไดอะแกรมของวงจรจะเป็นดังรูปที่ 11

วิธีนี้จึงเสมือนการใช้ IOP นั่งเฝ้าจอภาพแล้วนำเอาจอภาพมาจัดใหม่ให้ถูกต้อง



รูปที่ 11 ไดอะแกรมการใช้ IOP

จอภาพเดิมจะเป็นภาษาไทยแบบบรรทัดเดียว ซึ่งทำให้โปรแกรมสำเร็จรูปทั่วไปรับได้อยู่แล้วและ IOP จะจัดการให้เป็น 24 บรรทัด ภาษาไทยอย่างถูกอักขระ

สถานการณ์ในปัจจุบัน

ในปัจจุบันการพัฒนาการได้กระจายกันในแต่ละเทคนิค แต่ที่สำคัญคือจำนวนสแกนของเส้น ถ้าให้เส้นมาก(ตัวอักษรจะเล็กลง) แต่ตัวอักษรสวยขึ้นจะทำให้ภาพพริ้วที่เรียกว่า fricker เพราะจำนวนเฟรมลดลงโดยสังเกตจากแบบอินเตอร์เลซ ภาพจะเป็น 25 เฟรมต่อวินาที บนจอโมโนโครม จะเห็นภาพสั่นเล็กน้อย และถ้าเราเปลี่ยนจำนวนเส้นให้มากขึ้นโดยโปรแกรมบอก 6845 ภาพก็จะสั่นมากขึ้น เพราะจำนวนสัญญาณดอทคงที่ แต่ถ้าได้เส้นมากก็จะลดจำนวนเฟรมลง ปัญหาจึงอยู่ที่ว่าไม่มีใครออกแบบเองทั้งหมด จึงไม่ได้เพิ่มสัญญาณดอทให้สูงเกินกว่า 16 MHz (ความจริงอาจกำหนดได้สูงถึง 21 MHz) และที่สำคัญคือจอภาพคงมีขนาดใหญ่ขึ้นอีกเล็กน้อยเพื่อให้มองตัวอักษรสบายตาขึ้น

ปัญหามิได้อยู่ที่ฮาร์ดแวร์

ระบบฮาร์ดแวร์ในปัจจุบันสามารถเรียนรู้และเข้าใจได้ ผู้เขียนนำวงจรเท่าที่มีเกือบทุกระบบในเมืองไทย ลองมาอธิบายในชั้นเรียนให้นิสิตดู ก็ปรากฏว่านิสิตเข้าใจได้ไม่ยาก การดัดแปลงทางฮาร์ดแวร์จึงเป็นเรื่องที่ทำกันได้โดยไม่ยาก ถึงแม้จะออกแบบบอร์ดอะแดปเตอร์นี้ใหม่ก็ทำได้ โดยไม่ยากเช่นกัน

ปัญหาที่สำคัญอยู่ที่ซอฟท์แวร์ เพราะเราต้องการใช้ซอฟท์แวร์สำเร็จรูป เช่น โลตัส ดีเบส3 ฯลฯ เราไม่มีโอกาสรู้ว่าซอฟท์แวร์เหล่านั้นติดต่อกับอินพุทเอาท์พุทอย่างไร ดังนั้น TSX ที่จัดการกับอินพุทเอาท์พุทจึงไม่สามารถใช้ได้กับซอฟท์แวร์สำเร็จรูปทุกโปรแกรม

เมื่อเป็นเช่นนี้ ผู้พัฒนาจึงต้องเข้าไปแก้ไขภายในตัวโปรแกรมสำเร็จรูปกันทั้งสิ้น การเข้าไปแก้ไขโปรแกรมสำเร็จรูปนี้เป็นเรื่องใหญ่และหากซอฟท์แวร์นั้นเปลี่ยนเวอร์ชั่น นั่นแหละคือปัญหาที่ต้องแก้ไขกันต่อไปไม่รู้จักจบสิ้น และการแก้ไขนี้ถ้าทำต่อไปเรื่อยๆ จะไม่คุ้มค่า และต้องยุ่งเกี่ยวกับโปรแกรมสำเร็จรูปของต่างประเทศนับว่าไม่ดี

หากโปรแกรมเขียนขึ้นเองและพัฒนาต่อมา ปัญหาจะไม่ค่อยมี หากบำรุงรักษาให้ดี ดังนั้นผู้เขียนใคร่อยากให้มีการพัฒนาซอฟท์แวร์ของตัวเองมากขึ้น

อนาคตของไทย 25

ผู้เขียนอยากเห็นเครื่องพิมพ์ที่มีความฉลาดในตัวเอง ที่พิมพ์ภาษาไทยได้เองอย่างถูกต้องโดยไม่ต้องจัดระดับก่อนในคอมเตอร์ แต่ปรากฏว่ายังไม่มีใครทำ หรืออเห็นเครื่องพิมพ์ฯ 3 ระดับทีเดียว จะเพิ่มคเร็วขึ้นอีก 3 เท่าอย่างไม่ยาก อนาคตของ25 น่าจะทำให้เครื่องพิมพ์มีความฉลาดใครจะเป็นผู้เริ่มก่อน.....

ผู้เขียนอยากจะเห็นฮาร์ดแวร์โมโนไอะแดปเตอร์แบบ IOP ที่ออกแบบเอง โโปรเซสเซอร์ควบคุมการแสดงผล โดให้บอร์ดนี้มีความฉลาดในตัวเอง สามจัดระดับ ทำให้การทำงานของระบบไมคอมพิวเตอร์ทำงานเร็วขึ้น และลดปัการไปแก้ไขซอฟท์แวร์อันจะต้องผูกพันลิขสิทธิ์ของต่างประเทศ ซึ่งอาจมีปัญหาภายหลัง แต่หากใช้ IOP แล้วจะทำให้ไม่ยุ่งเกี่ยว เพราะบอร์ดจะมีความฉลาดในเฝ้าจอภาพ (วิดีโอแรมเดิม) แล้วนำมาใหม่ให้เราอย่างชาญฉลาดตามความต้อง....ใครจะเป็นผู้เริ่มทำ

ผู้เขียนอยากเห็นการพัฒนาซอฟท์บนบอร์ดไทย 25 ในรูปแบบโปรแกรมสำรูปทั่วไป นอกจากเวิร์ดโปรเซสซิ่ง สเปรดชีท ดาต้าเบส กราฟิก ทำให้เซอฟท์แวร์บางอย่างใช้งานได้ และพัความสามารถของทรัพยากรบุคคลให้ขึ้น เราน่าเริ่มได้แล้ว.....

ความหวังยังแจ่มใส.....